

```
1 <?php
2
3 /**
4  * @author GOLAY Brian
5  * @version 1.0 (2021/05/20)
6  * Flight-related functions
7  */
8
9 namespace FlightClub\sql;
10
11 use FlightClub\sql\DBConnection;
12
13 require_once 'dbConnection.php';
14
15 /**
16  * Flight related class that contain flight related functions with database
17  * interaction
18  */
19 class FlightDAO
20 {
21     /**
22      * Get all flight of an user with his id
23      *
24      * @param int $idUser
25      * @return array[mixed]
26      */
27     public static function getAllUserFlightByUserId($idUser)
28     {
29         $db = DBConnection::getConnection();
30         $sql = "SELECT * FROM `tbl_flight` join `tbl_user-flight` on
31 `tbl_flight`.`Id_Flight` = `tbl_user-flight`.`Id_Flight` WHERE `Id_User` = :idUser";
32         $request = $db->prepare($sql);
33         $request->execute([
34             ':idUser' => $idUser
35         ]);
36         return $request->fetchAll();
37     }
38
39     /**
40      * Get all flight of an user with his id with only the data needed for the PDF
41      *
42      * @param int $idUser
43      * @return array[mixed]
44      */
45     public static function getAllUserFlightByUserIdForExport($idUser)
46     {
47         $db = DBConnection::getConnection();
48         $sql = "SELECT `Cd_Role`, `No_Flight`, `Dttm_Departure`, `Dttm_Arrival`,
49 `Tm_Engine_On`, `Tm_Engine_Off`, `Nm_Plane`, `No_Plane`, `Cd_ICAO_Departure`,
50 `Cd_ICAO_Arrival`, `Cd_Flight_Type`, `Cd_Flight_Mode`, `Nb_Passengers` FROM
51 `tbl_flight` join `tbl_user-flight` on `tbl_flight`.`Id_Flight` = `tbl_user-
52 flight`.`Id_Flight` WHERE `Id_User` = :idUser";
53         $request = $db->prepare($sql);
54         $request->execute([
55             ':idUser' => $idUser
56         ]);
57         return $request->fetchAll();
58     }
59 }
```

```

55  /**
56  * Add a flight to the database
57  *
58  * @param int $idFlight
59  * @param int $idUser
60  * @param string $role
61  * @param string $flight
62  * @param string $dateDeparture
63  * @param string $dateArrival
64  * @param string $timeDeparture
65  * @param string $timeArrival
66  * @param string $timeEngineOn
67  * @param string $timeEngineOff
68  * @param string $typeAircraft
69  * @param string $registrationPlane
70  * @param string $icaoDeparture
71  * @param string $icaoArrival
72  * @param string $flightType
73  * @param string $flightMode
74  * @param string $weather
75  * @param int $passengers
76  * @param string $notes
77  * @return void
78  */
79  public static function addFlight(
80      $idFlight,
81      $idUser,
82      $role,
83      $flight,
84      $dateDeparture,
85      $dateArrival,
86      $timeEngineOn,
87      $timeEngineOff,
88      $typeAircraft,
89      $registrationPlane,
90      $icaoDeparture,
91      $icaoArrival,
92      $flightType,
93      $flightMode,
94      $weather,
95      $passengers,
96      $notes
97  ) {
98
99      $db = DBConnection::getConnection();
100      $sql = "INSERT INTO `tbl_flight`(`Id_Flight`,
`No_Flight`, `Dttm_Departure`, `Dttm_Arrival`,
`Tm_Engine_On`, `Tm_Engine_Off`, `Nm_Plane`, `No_Plane`, `Cd_ICAO_Departure`,
`Cd_ICAO_Arrival`, `Cd_Flight_Type`, `Cd_Flight_Mode`, `Txt_Meteo`, `Nb_Passengers`, `Txt_Note`)
101      VALUES (:idFlight, :flight, :dateDeparture, :dateArrival, :timeEngineOn,
102          :timeEngineOff, :typeAircraft, :registrationPlane, :icaoDeparture,
:icaoArrival, :flightType, :flightMode, :weather, :passengers, :notes);
103      INSERT INTO `tbl_user-flight`(`Id_User`, `Id_Flight`, `Cd_Role`)
104      VALUES (:idUser, :idFlight, :role)";
105      $request = $db->prepare($sql);
106      $request->execute([
107          ':idFlight' => $idFlight,
108          ':idUser' => $idUser,

```

```

111         ':role' => $role,
112         ':flight' => $flight,
113         ':dateDeparture' => $dateDeparture,
114         ':dateArrival' => $dateArrival,
115         ':timeEngineOn' => $timeEngineOn,
116         ':timeEngineOff' => $timeEngineOff,
117         ':typeAircraft' => $typeAircraft,
118         ':registrationPlane' => $registrationPlane,
119         ':icaoDeparture' => $icaoDeparture,
120         ':icaoArrival' => $icaoArrival,
121         ':flightType' => $flightType,
122         ':flightMode' => $flightMode,
123         ':weather' => $weather,
124         ':passengers' => $passengers,
125         ':notes' => $notes
126     ]);
127 }
128
129
130 /**
131  * Edit a flight to the database
132  *
133  * @param int $idFlight
134  * @param int $idUser
135  * @param string $role
136  * @param string $flight
137  * @param string $dateDeparture
138  * @param string $dateArrival
139  * @param string $timeEngineOn
140  * @param string $timeEngineOff
141  * @param string $typeAircraft
142  * @param string $registrationPlane
143  * @param string $icaoDeparture
144  * @param string $icaoArrival
145  * @param string $flightType
146  * @param string $flightMode
147  * @param string $weather
148  * @param int $passengers
149  * @param string $notes
150  * @return void
151  */
152 public static function editFlight(
153     $idFlight,
154     $idUser,
155     $role,
156     $flight,
157     $dateDeparture,
158     $dateArrival,
159     $timeEngineOn,
160     $timeEngineOff,
161     $typeAircraft,
162     $registrationPlane,
163     $icaoDeparture,
164     $icaoArrival,
165     $flightType,
166     $flightMode,
167     $weather,
168     $passengers,
169     $notes
170 ) {

```

```

171
172     $db = DBConnection::getConnection();
173     $sql = "UPDATE `tbl_flight` SET `No_Flight` = :flight, `Dttm_Departure` =
:dateDeparture, `Dttm_Arrival` = :dateArrival,
174     `Tm_Engine_On` = :timeEngineOn, `Tm_Engine_Off` = :timeEngineOff, `Nm_Plane` =
:typeAircraft, `No_Plane` = :registrationPlane, `Cd_ICAO_Departure` = :icaoDeparture,
175     `Cd_ICAO_Arrival` = :icaoArrival, `Cd_Flight_Type` =
:flightType, `Cd_Flight_Mode` = :flightMode, `Txt_Meteo` = :weather, `Nb_Passengers` =
:passengers, `Txt_Note` = :notes WHERE `tbl_flight`.`Id_Flight` = :idFlight;
176     UPDATE `tbl_user-flight` SET `Cd_Role` = :role WHERE `Id_Flight` =
:idFlight)";
177     $request = $db->prepare($sql);
178     $request->execute([
179         ':idFlight' => $idFlight,
180         ':idUser' => $idUser,
181         ':role' => $role,
182         ':flight' => $flight,
183         ':dateDeparture' => $dateDeparture,
184         ':dateArrival' => $dateArrival,
185         ':timeEngineOn' => $timeEngineOn,
186         ':timeEngineOff' => $timeEngineOff,
187         ':typeAircraft' => $typeAircraft,
188         ':registrationPlane' => $registrationPlane,
189         ':icaoDeparture' => $icaoDeparture,
190         ':icaoArrival' => $icaoArrival,
191         ':flightType' => $flightType,
192         ':flightMode' => $flightMode,
193         ':weather' => $weather,
194         ':passengers' => $passengers,
195         ':notes' => $notes
196     ]);
197 }
198
199
200
201 /**
202  * Get all flight order by their id descending
203  *
204  * @return array[mixed]
205  */
206 public static function getAllFlightOrderById()
207 {
208     $db = DBConnection::getConnection();
209     $sql = "SELECT * FROM `tbl_flight` ORDER BY `Id_Flight` desc";
210     $request = $db->prepare($sql);
211     $request->execute();
212     return $request->fetchAll();
213 }
214
215
216 /**
217  * Get a flight with his Id
218  *
219  * @param int $idFlight
220  * @return array[mixed]
221  */
222 public static function getFlightById($idFlight)
223 {
224     $db = DBConnection::getConnection();
225     $sql = "SELECT * FROM `tbl_flight` WHERE `Id_Flight` = :id";

```

```
226     $request = $db->prepare($sql);
227     $request->execute([
228         ':id' => $idFlight
229     ]);
230     return $request->fetch();
231 }
232
233 }
234
```